IN THE CLAIMS:

1. (Previously Presented) A method for distributing data over a wide area network, comprising:

receiving at a master, coupled to a plurality of slaves, a slave state of a first slave of said plurality of slaves, wherein the slave state of the first slave includes a current file length, for each of a plurality of files stored at the first slave, the current file length of each respective file representing an amount of the respective file currently stored at the first slave;

determining at said master that said first slave requires data;

selecting at said master, in accordance with a predefined bandwidth resource allocation method, a second slave of said plurality of slaves from which to transfer said data to said first slave;

scheduling at said master, a data transfer of said data from said second slave to said first slave;

transmitting instructions from said master to said first slave, instructing said first slave to obtain said data from said second slave; and

after transmitting the instructions, receiving from said first slave a revised state of said first slave, and subsequently repeating the determining, scheduling and transmitting operations.

2. (Previously Presented) The method of claim 1, wherein said slave state of said first slave is a saved current state of file storage of said first slave including an indication of files that said first slave requires and an indication of what portion, if any, of a particular one of the files that said first slave has already stored in memory.

3. (Previously Presented) The method of claim 1, wherein said selecting said second slave further comprises:

determining which available slaves of said plurality of slaves have said data;

ascertaining system resources; and

establishing that said second slave is a preferred slave, from said available slaves, from which to transfer said data to said first slave, based on said system resources.

4. (Previously Presented) The method of claim 3, wherein said ascertaining further comprises:

determining differences between maximum system resources of the system and resources in use.

5. (Previously Presented) The method of claim 1, wherein said scheduling further comprises:

determining overall system resources; and

establishing that said second slave is a preferred slave from which to transfer said data to said first slave based on said overall system resources.

6. (Previously Presented) The method of claim 1, wherein said scheduling further comprises:

determining incremental system resources that have changed since last determining said incremental system resources; and

establishing that said second slave is a preferred slave from which to transfer said data to said first slave based on said incremental system resources.

7. (Previously Presented) The method of claim 1, wherein said scheduling further comprises:

determining partition system resources of a partition of system components that includes said first slave; and

establishing that said second slave is a preferred slave from which to transfer said data to said first slave based on said partition system resources.

8. (Original) The method of claim 1, further comprising, at any time before said scheduling:

determining at said master, overall system resources;

computing at said master, a preferred data transfer schedule based on said overall system resources;

storing said data transfer schedule on said master, such that said scheduling further comprises selecting said instructions from said data schedule.

9. (Previously Presented) A method for distributing data over a wide area network, comprising:

transmitting a file set from a master, coupled to a plurality of slaves, to a first slave of a plurality of slaves, where said file set is a list of files and version numbers of said files that said first slave must obtain, receiving at said master a slave state of said first slave of said plurality of slaves, wherein the slave state of the first slave includes a current file length, for each of the files in the list of files in said file set, the current file length of each respective file representing an amount of the respective file currently stored at the first slave;

determining at said master that said first slave requires data;

selecting at said master, in accordance with a predefined bandwidth resource allocation method, a second slave of said plurality of slaves from which to transfer said data to said first slave;

scheduling at said master, a data transfer of said data from said second slave to said first slave; and

transmitting instructions from said master to said first slave, instructing said first slave to obtain said data from said second slave.

10. (Original) The method of claim 9, further comprising, after said obtaining:

ascertaining at said first slave, said state of said first slave; and

transmitting said state to said master.

11. (Previously Presented) The method of claim 10, wherein said state is ascertained by establishing what data said first slave still requires.

12. (Original) The method of claim 11, wherein said establishing further comprises subtracting data already stored on said first slave from that required per said file set.

13. (Previously Presented) The method of claim 1, further comprising:

receiving said instructions at said first slave;

requesting said data from said second slave;

receiving at said first slave said data from said second slave; and

storing said data on said first slave.

14. (Canceled)

15. (Previously Presented) The method of claim 13, wherein said ascertaining further comprises determining whether said transfer of said data from said second slave to said first slave was successful.

16. (Previously Presented) The method of claim 15, wherein if said transfer of said data was unsuccessful:

assessing at said first slave failure information identifying components of said WAN responsible for said unsuccessful transfer; and storing said failure information in said state, such that said failure information is transferred to said master.

17. (Original) The method of claim 16, further comprising:

incrementing a count for each component of said WAN that has been identified as being responsible for an unsuccessful transfer;

identifying a component of said WAN as failed when said count reaches a predetermined number.

18. (Previously Presented) The method of claim 13, further comprising verifying said data.

19. (Original) The method of claim 13, further comprising:

determining that no more data is required for a particular file; and

updating said state to show that no more data is required for said particular file.

20. (Previously Presented) The method of claim 1, further comprising, before said receiving:

generating said data at a generator;

generating a checksum of said data at said generator; and

sending said data and said checksum to said second slave.

21. (Previously Presented) The method of claim 20, further comprising, before said sending:

generating a final flag at said generator, where said final flag indicates that said data is a final block of data for a particular file, such that said final flag is sent to said second slave together with said data and said checksum.

22. (Previously Presented) The method of claim 21, further comprising:

receiving said data at said second slave; and

storing said data on said second slave.

23. (Previously Presented) The method of claim 22, further comprising, after said second slave receives said data, verifying said data based on said checksum.

24. (Previously Presented) A system for distributing data, comprising:

a plurality of dispersed slaves configured to store and serve data;

at least one generator configured to generate said data and to make said data available to a subset of said plurality of slaves; and a master coupled to said plurality of dispersed slaves, where said master is configured to:

transmit a file set to a first slave of a plurality of slaves, where said file set is a list of files and version numbers of said files that said first slave must obtain;

receive state information from a first slave of said plurality of slaves, wherein the state information from the first slave includes a current file length, for each of the files in the list of files in said file set, the current file length of each respective file representing an amount of the respective file currently stored at the first slave;

determine that said first slave requires data;

select, in accordance with a predefined bandwidth resource allocation method, a second slave of said plurality of slaves from which to transfer said data to said first slave;

schedule a data transfer of said data from said second slave to said first slave; and

transmit instructions from said master to said first slave, instructing said first slave to obtain said data from said second slave.

25. (Previously Presented) The system of claim 24, wherein said master comprises:

a processor;

communications circuitry; and

a memory, comprising:

an operating system;

communication procedures for communicating with said plurality of slaves via said communication circuitry;

an indication maximum of system resources;

an indication of system resources in use;

an indication of a state of each slave; and

a scheduler for scheduling data transfer between slaves.

26. (Previously Presented) The system of claim 25, wherein each slave of said slaves further comprises:

a processor;

communications circuitry; and

a memory, comprising:

an operating system;

communication procedures for communicating with said plurality of slaves and said master via said communication circuitry;

a file set file containing a list of files required by said slave; and

at least a portion of one of said files and an indication of how much of said file is still required.

27. (Previously Presented) The system of claim 25, wherein said master and each of said slaves include verification procedures for verifying said data.

28. (Original) A method for distributing a plurality of files over a network, comprising:

at each slave of a plurality of slaves, determining a current file length for each of the plurality of files, the current file length of each respective file representing an amount of the respective file currently stored at the slave, and sending slave status information to a master, the slave status information including the current file length for each of the plurality of files;

at the master, receiving the slave status information from each of the slaves;

at the master, scheduling copy operations based on the slave status information received from the slaves, each scheduled copy operation for copying a specified portion of a specified file of the plurality of files, from a first specified slave of the plurality of slaves to a second specified slave of the plurality of slaves; and

at the master, storing bandwidth capability information indicating data transmission bandwidth capabilities for a plurality of resources required for transmitting data between the plurality of slaves, and storing bandwidth usage information indicating amounts of data transmission bandwidth allocated to copy operations scheduled by the master, the stored bandwidth usage information indicating a total allocated bandwidth for each resource of the plurality of resources;

the scheduling including allocating for each copy operation an amount of data transmission bandwidth with respect to each resource required by the copy operation, and updating the stored bandwidth usage information in accordance therewith;

the scheduling further including only scheduling copy operations that do not cause the total allocated bandwidth of any resource to exceed the bandwidth capability of that resource.

29. (Original) The method of claim 28, wherein a first file of the plurality of files includes a first set of blocks and a second set of blocks, the method including:

executing the copy operations scheduled by the master; and at one of the slaves, generating the second set of blocks of the first file after at least one block of the first file has been copied to another one of the slaves by the execution of a respective copy operation scheduled by the master.

30. (Original) The method of claim 28, wherein the scheduling includes applying a cost function to a plurality of potential copy operations to produce a respective score for each potential copy operation, selecting a copy operation of the plurality of potential copy operations having a best score, and scheduling the selected copy operation.

31. (Previously Presented) The method of claim 9, wherein said slave state of said first slave is a saved current state of file storage of said first slave including an indication of files that said first slave requires and an indication of what portion, if any, of a particular one of the files that said first slave has already stored in memory.

32. (Previously Presented) The method of claim 9, wherein said selecting said second slave comprises:

determining which available slaves of said plurality of slaves have said data;

ascertaining system resources; and

establishing that said second slave is a preferred slave, from said available slaves, from which to transfer said data to said first slave, based on said system resources.

33. (Previously Presented) The method of claim 32, wherein said ascertaining further comprises:

determining differences between maximum system resources of the system and resources in use.

34. (Previously Presented) The method of claim 9, wherein said scheduling further comprises:

determining overall system resources; and

establishing that said second slave is a preferred slave from which to transfer said data to said first slave based on said overall system resources.

35. (Previously Presented) The method of claim 9, wherein said scheduling further comprises:

determining incremental system resources that have changed since last determining said incremental system resources; and

establishing that said second slave is a preferred slave from which to transfer said data to said first slave based on said incremental system resources.

36. (Previously Presented) The method of claim 9, wherein said scheduling further comprises:

determining partition system resources of a partition of system components that includes said first slave; and

establishing that said second slave is a preferred slave from which to transfer said data to said first slave based on said partition system resources.

37. (Previously Presented) The method of claim 9, further comprising, at any time before said scheduling:

determining at said master, overall system resources;

computing at said master, a preferred data transfer schedule based on said overall system resources;

storing said data transfer schedule on said master, such that said scheduling further comprises selecting said instructions from said data schedule.

38. (Previously Presented) A system for distributing data, comprising:

a plurality of dispersed slaves configured to store and serve data;

at least one generator configured to generate said data and to make said data available to a subset of said plurality of slaves; and

a master coupled to said plurality of dispersed slaves, where said master is configured to:

receive state information from a first slave of said plurality of slaves, wherein the slave information from the first slave includes a current file length, for each of a plurality of files stored at the first slave, the current file length of each respective file representing an amount of the respective file currently stored at the first slave;

determine that said first slave requires data;

select, in accordance with a predefined bandwidth resource allocation method, a second slave of said plurality of slaves from which to transfer said data to said first slave;

schedule a data transfer of said data from said second slave to said first slave;

transmit instructions from said master to said first slave, instructing said first slave to obtain said data from said second slave; and

after transmitting the instructions, receive from said first slave a revised state of said first slave, and subsequently repeat the determining, scheduling and transmitting operations.

39. (Previously Presented) A computer readable storage medium storing one or more programs for execution by one or more processors of a master system so as distribute data among a plurality slaves, the one or more programs comprising instructions for:

receiving at the master system a slave state of a first slave of said plurality of slaves, wherein the slave state of the first slave includes a current file length, for each of a plurality of files stored at the first slave, the current file length of each respective file representing an amount of the respective file currently stored at the first slave;

determining, at the master system, that said first slave requires data;

selecting, in accordance with a predefined bandwidth resource allocation method, a second slave of said plurality of slaves from which to transfer said data to said first slave;

scheduling a data transfer of said data from said second slave to said first slave;

transmitting instructions from said master system to said first slave, instructing said first slave to obtain said data from said second slave; and

after transmitting the instructions, receiving from said first slave a revised state of said first slave, and subsequently repeating the determining, scheduling and transmitting operations.

40. (Previously Presented) The computer readable storage medium of claim 39, wherein said slave state of said first slave is a saved current state of file storage of said first slave including an indication of files that said first slave requires and an indication of what portion, if any, of a particular one of the files that said first slave has already stored in memory.

41. (Previously Presented) The computer readable storage medium of claim 39, wherein said instructions for selecting said second slave include instructions for:

determining which available slaves of said plurality of slaves have said data;

ascertaining system resources; and

establishing that said second slave is a preferred slave, from said available slaves, from which to transfer said data to said first slave, based on said system resources.

42. (Previously Presented) The computer readable storage medium of claim 39, the one or more programs further including instructions for:

determining at said master, overall system resources;

computing at said master, a preferred data transfer schedule based on said overall system resources; and storing said data transfer schedule on said master, such that said instructions for scheduling include instructions for selecting from said data schedule said instructions to transmit to said first slave.

43. (Previously Presented) A computer readable storage medium storing one or more programs for execution by one or more processors of a master system so as distribute data among a plurality slaves, the one or more programs comprising instructions for:

transmitting a file set from the master system to a first slave of a plurality of slaves, where said file set is a list of files and version numbers of said files that said first slave must obtain,

receiving at said master system a slave state of said first slave of said plurality of slaves, wherein the slave state of the first slave includes a current file length, for each of the files in the list of files in said file set, the current file length of each respective file representing an amount of the respective file currently stored at the first slave;

determining at said master system, that said first slave requires data;

selecting at said master system, in accordance with a predefined bandwidth resource allocation method, a second slave of said plurality of slaves from which to transfer said data to said first slave;

scheduling a data transfer of said data from said second slave to said first slave; and

transmitting instructions from said master system to said first slave, instructing said first slave to obtain said data from said second slave.